UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/687,248	10/16/2003	Jimmie Earl DeWitt JR.	AUS920030547US1	9397

35525 7590 01/06/2006

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
LEE, CHRISTOPHER E	

ART UNIT	PAPER NUMBER
2112	

DATE MAILED: 01/06/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/687,248	DEWITT ET AL.	
	Examiner	Art Unit	
	Christopher E. Lee	2112	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *16 October 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *10/16/03*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Specification

1. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

Double Patenting

2. Claims 1 and 3-8 of this application conflict with claims 1-5, 7, and 9 of Application No. 10/704,117. 37 CFR 1.78(b) provides that when two or more applications filed by the same applicant contain conflicting claims, elimination of such claims from all but one application may be required in the absence of good and sufficient reason for their retention during pendency in more than one application. Applicant is required to either cancel the conflicting claims from all but one application or maintain a clear line of demarcation between the applications. See MPEP § 822.

The U.S. Patent and Trademark Office normally will not institute an interference between applications or a patent and an application of common ownership (see MPEP § 2302). Commonly assigned Application No. 10/704,117, discussed above, would form the basis for a rejection of the noted claims under 35 U.S.C. 103(a) if the commonly assigned case qualifies as prior art under 35 U.S.C. 102(e), (f) or (g) and the conflicting inventions were not commonly owned at the time the invention in this application was made. In order for the examiner to resolve this issue, the assignee can, under 35 U.S.C. 103(c) and 37 CFR 1.78(c), either show that the conflicting inventions were commonly owned at the time the invention in this application was made, or name the prior inventor of the conflicting subject matter.

A showing that the inventions were commonly owned at the time the invention in this application was made will preclude a rejection under 35 U.S.C. 103(a) based upon the commonly assigned case as a

reference under 35 U.S.C. 102(f) or (g), or 35 U.S.C. 102(e) for applications pending on or after December 10, 2004.

3. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

4. Claims 1, 3-8, 10, 12-18, and 20-22 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-5, 7, 9, 11-15, and 17-20 of copending Application No. 10/704,117 in view of Torrey et al. [US 6,145,123 A; hereinafter Torrey]. Although the conflicting claims are not identical, they are not patentably distinct from each other because the instant application is anticipated by the copending application in view of Torrey in that the copending application in view of Torrey suggests obviously all the limitations of the instant application. Therefore, the instant application is not patentably distinct from the earlier filed claims of the copending application in view of Torrey and as such is unpatentable for obvious-type double patenting.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

In regard to claim 1, the copending application discloses all the same limitations as the limitations of the claim 1 in the copending application with the exception of the limitation "said processor being in the data processing system, said indicator indicates enabling a mode of operation in which interrupts are to be generated; and responsive to receiving a subsequent instruction after receiving the instruction."

However, Torrey discloses an information processing system with trace on/off control (See Abstract), wherein responsive to receiving an instruction for execution in an instruction cache (i.e., Core 112 of Fig. 1) in a processor (i.e., Processor 110 of Fig. 1) being in a data processing system (i.e., information processing system 600 of Fig. 6; See col. 5, lines 9-11), determining whether the instruction indicates enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19); and responsive to receiving a subsequent instruction (i.e., next instruction in Step 530 of Fig. 5) after receiving the instruction (See col. 6, lines 56-61).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said trace on/off control, as disclosed by Torrey, in said data processing system, as disclosed by the instant application, for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

In regard to claim 3, the copending application teaches all the same limitations as the limitations of the claim 2 in the copending application.

In regard to claim 4, the copending application teaches all the same limitations as the limitations of the claim 3 in the copending application.

In regard to claim 5, the copending application teaches all the same limitations as the limitations of the claim 4 in the copending application.

In regard to claim 6, the copending application teaches all the same limitations as the limitations of the claim 5 in the copending application.

In regard to claim 10, the copending application discloses all the same limitations as the limitations of the claim 11 in the copending application with the exception of the limitation "said processor being in the data processing system, said indicator indicates enabling a mode of operation in

which interrupts are to be generated; and second determining means, responsive to receiving a subsequent instruction after receiving the instruction."

However, Torrey discloses an information processing system with trace on/off control (See Abstract), wherein first determining means (i.e., means for controlling trace on/off in the flow in Fig. 5), responsive to receiving an instruction for execution in an instruction cache (i.e., Core 112 of Fig. 1) in a processor (i.e., Processor 110 of Fig. 1) being in a data processing system (i.e., information processing system 600 of Fig. 6; See col. 5, lines 9-11), for determining whether the instruction indicates enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19); and second determining means (i.e., means for tracing instruction), responsive to receiving a subsequent instruction (i.e., next instruction in Step 530 of Fig. 5) after receiving the instruction (See col. 6, lines 56-61).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said trace on/off control, as disclosed by Torrey, in said data processing system, as disclosed by the instant application, for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

In regard to claim 12, the copending application teaches all the same limitations as the limitations of the claim 12 in the copending application.

In regard to claims 7 and 16, the copending application teaches all the same limitations as the limitations of the claim 7 in the copending application.

In regard to claims 8 and 17, the copending application teaches all the same limitations as the limitations of the claim 9 in the copending application.

In regard to claim 18, the copending application discloses all the same limitations as the limitations of the claim 17 in the copending application with the exception of the limitation "said processor being in the data processing system, said indicator indicates enabling a mode of operation in which interrupts are to be generated; and second instructions, responsive to receiving a subsequent instruction after receiving the instruction."

However, Torrey discloses an information processing system with trace on/off control (See Abstract), wherein first instructions (i.e., controlling trace on/off in the flow in Fig. 5), responsive to receiving an instruction for execution in an instruction cache (i.e., Core 112 of Fig. 1) in a processor (i.e., Processor 110 of Fig. 1) being in a data processing system (i.e., information processing system 600 of Fig. 6; See col. 5, lines 9-11), for determining whether the instruction indicates enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19); and second instructions (i.e., tracing instructions), responsive to receiving a subsequent instruction (i.e., next instruction in Step 530 of Fig. 5) after receiving the instruction (See col. 6, lines 56-61).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have implemented said trace on/off control, as disclosed by Torrey, on said computer program product, as disclosed by the instant application, for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

In regard to claim 20, the copending application teaches all the same limitations as the limitations of the claim 18 in the copending application.

In regard to claim 21, the copending application teaches all the same limitations as the limitations of the claim 19 in the copending application.

In regard to claim 22, the copending application teaches all the same limitations as the limitations of the claim 20 in the copending application.

5. The mapping of the rejected claims in the instant application to the copending application is as follows.

Instant Application 10/687,248	copending Application No. 10/704,117
1	1 + Torrey
3	2
4	3
5	4
6	5
7, 16	7
8, 17	9
10	11 + Torrey
12	12
13	13
14	14
15	15
18	17 + Torrey
20	18
21	19
22	20

Claim Objections

6. The claim 1 recites the subject matter "the instruction" in lines 9-10. However, it has not been specifically clarified if it is the subject matter "an instruction" in line 3 or the subject matter "a subsequent instruction" in line 8. Therefore, the Examiner presumes that the term "the instruction" in lines 9-10 should be considered as --the subsequent instruction-- in light of the specification since the subject matter "an instruction" in line 3 could not be of a certain type, but indicates enabling a mode of operation in which interrupts are to be generated.

7. The claims 7 and 16 recite the subject matter "the indicator" in line 2 of the claim 7, and in lines 2-3 of the claim 16, respectively. However, it has not been specifically clarified in the claims 7 and 16, and their intervening claims, respectively. Therefore, the Examiner presumes that the term "the indicator" could be considered as --an indicator-- in light of the specification since it is not defined in the claims.

8. The claim 9 recites the subject matter "the interrupt" in line 2. However, it has not been specifically clarified in the claim 9. Therefore, the Examiner presumes that the term "the interrupt" could be considered as --the interrupt unit-- in light of the specification since it is not defined in the claim.

9. The claim 18 recites the subject matter "the data processing system" in line 6. However, it has not been specifically clarified in the claim 18. Therefore, the Examiner presumes that the term "the data processing system" could be considered as --a data processing system-- in light of the specification since it is not defined in the claim.

Claim Rejections - 35 USC § 102

10. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

11. Claims 1, 3-6, 8-10, 12-15, 17, 18, and 20-22 are rejected under 35 U.S.C. 102(b) as being anticipated by Smolders [US 6,253,338 B1].

Referring to claim 1, Smolders discloses a method in a data processing system for processing instructions (i.e., method within a data processing system for counting various events from a running program; See Abstract), the method comprising:

- responsive to receiving an instruction for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the instruction (i.e., instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11);

- responsive to receiving a subsequent instruction after receiving the instruction (i.e., said instruction for setting BE), determining whether the subsequent instruction is of a certain type (i.e., checking for branch instructions; See col. 3, line 67 through col. 4, line 6); and
- generating an interrupt (i.e., trace interrupt) if the mode of operation in which interrupts are to be enabled and the instruction is of the certain type (See col. 3, lines 58-61).

Referring to claim 3, Smolders teaches the generating step (See col. 3, lines 58-61) comprising

- sending a signal (i.e., trace interrupt signal after Branch instruction at Step 30 in Fig. 3) from an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in Fig. 2) to an interrupt unit (i.e., Performance monitor 24 of Fig. 2) in the processor (i.e., Processor 12 of Fig. 2; See col. 4, lines 21-26); and
- processing the interrupt (i.e., branch tracing interrupt) in the interrupt unit (i.e., said Performance monitor; in fact, counter level tracing tool 31 in Fig. 3 performs handling said branch tracing interrupt) in response to receiving the signal at the interrupt unit (See Fig. 3 and col. 4, lines 12+).

Referring to claim 4, Smolders teaches the processing step (See Fig. 3 and col. 4, lines 12+) including

- executing code associated with the interrupt (i.e., performing interrupt handling for branch tracing interrupt; See col. 3, lines 55-61, wherein in fact, the interrupt handler code starts executing and the instruction flow unit is programmed to generate a trace interrupt after each branch inherently anticipates the claimed limitation "executing code associated with the interrupt").

Referring to claim 5, Smolders teaches

- the code records cache misses by a functional unit (i.e., load/store unit) attempting to access instructions in a cache (i.e., L2 Cache; See col. 3, lines 43-44).

Referring to claim 6. Smolders teaches

- the code counts a number of times (i.e., counting selected events, e.g., branch tracing interrupt) the instruction of the certain type (i.e., branch instruction) has been executed (See col. 4, lines 6-11).

Referring to claim 8, Smolders teaches

- the instruction of the certain type is a branch instruction (See col. 3, line 67 through col. 4, line 6).

Referring to claim 9. Smolders discloses a data processing system (i.e., Data Processing System 10 in Fig. 1) comprising:

- an interrupt unit (i.e., Performance monitor 24 of Fig. 2), wherein
  - the interrupt unit (i.e., said Performance monitor) executes code (i.e., counter level tracing tool 31 in Fig. 3) in response to receiving a signal is received (See col. 4, lines 15-17; i.e., in response to signaling after each branch instruction); and
- an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in Fig. 2), wherein
  - the instruction cache receives instructions (See col. 3, lines 4-13) and sends the signal to the interrupt unit (i.e., said signaling after each branch instruction) if a mode of operation to generate interrupts is enabled (i.e., a specified branch trace enable bit 80 in the machine state register 76 being set in Fig. 2, in other words, branch tracing mode is enabled by setting BE

bit in MSR) and an instruction of a certain type is received for execution by the instruction cache (See col. 3, line 58 through col. 4, line 11).

Referring to claim 10, Smolders discloses a data processing system for processing instructions (i.e., a data processing system for counting various events from a running program; See Abstract), the data processing system comprising:

- first determining means (i.e., means for checking Machine State Register 76 of Fig. 2), responsive to receiving an instruction for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the instruction (i.e., instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11);
- second determining means (i.e., means for finding branch instruction), responsive to receiving a subsequent instruction after receiving the instruction (i.e., said instruction for setting BE), determining whether the subsequent instruction is of a certain type (i.e., checking for branch instructions; See col. 3, line 67 through col. 4, line 6); and
- generating means (i.e., Instruction flow unit 26 of Fig. 2) for generating an interrupt (i.e., trace interrupt) if the mode of operation in which interrupts are to be enabled and the instruction is of the certain type (See col. 3, lines 58-61).

Referring to claim 12, Smolders teaches the generating means (See col. 3, lines 58-61) comprising

- means for sending a signal (i.e., Instruction flow unit 26 of Fig. 2 sending trace interrupt signal after Branch instruction at Step 30 of Fig. 3) from an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in Fig. 2) to an interrupt unit (i.e., Performance monitor 24 of Fig. 2) in the processor (i.e., Processor 12 of Fig. 2; See col. 4, lines 21-26); and
- means for processing the interrupt (i.e., counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) in the interrupt unit (i.e., said Performance monitor; in fact, counter level tracing tool 31 in Fig. 3 performs handling said branch tracing interrupt) in response to receiving the signal at the interrupt unit (See Fig. 3 and col. 4, lines 12+).

Referring to claim 13, Smolders teaches the processing means (i.e., counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) including

- executing means for executing code associated with the interrupt (i.e., means for performing interrupt handling for branch tracing interrupt; See col. 3, lines 55-61, wherein in fact, the interrupt handler code starts executing and the instruction flow unit is programmed to generate a trace interrupt after each branch inherently anticipates the claimed limitation "executing means for executing code associated with the interrupt").

Referring to claim 14, Smolders teaches

- the code records cache misses by a functional unit (i.e., load/store unit) attempting to access instructions in a cache (i.e., L2 Cache; See col. 3, lines 43-44).

Referring to claim 15, Smolders teaches

- the code counts a number of times (i.e., counting selected events, e.g., branch tracing interrupt) the instruction of the certain type (i.e., branch instruction) has been executed (See col. 4, lines 6-11).

Referring to claim 17, Smolders teaches

- the instruction of the certain type is a branch instruction (See col. 3, line 67 through col. 4, line 6).

Referring to claim 18, Smolders discloses a computer program product in a computer readable medium (See col. 6, lines 14-25) for processing instructions (i.e., a computer program product residing on a computer usable medium for providing counter level tracing for an information handling system; See col. 8, lines 39-41), the computer program product comprising:

- first instructions (i.e., instructions for checking Machine State Register 76 in Fig. 2), responsive to receiving an instruction for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in a data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the instruction (i.e., instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11);
- second instructions (i.e., instructions for finding branch instruction), responsive to receiving a subsequent instruction after receiving the instruction (i.e., said instruction for setting BE), determining whether the subsequent instruction is of a certain type (i.e., checking for branch instructions; See col. 3, line 67 through col. 4, line 6); and

- third instructions (i.e., instructions for controlling Instruction flow unit 26 in Fig. 2), for generating an interrupt (i.e., trace interrupt) if the mode of operation in which interrupts are to be enabled and the instruction is of the certain type (See col. 3, lines 58-61).

Referring to claim 20, Smolders teaches the third instructions (See col. 3, lines 58-61) comprising

- first sub-instructions for sending a signal (i.e., control instructions for Instruction flow unit 26 of Fig. 2 sending trace interrupt signal after Branch instruction at Step 30 of Fig. 3) from an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in Fig. 2) to an interrupt unit (i.e., Performance monitor 24 of Fig. 2) in the processor (i.e., Processor 12 of Fig. 2; See col. 4, lines 21-26); and
- second sub-instructions for processing the interrupt (i.e., instructions for counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) in the interrupt unit (i.e., said Performance monitor; in fact, counter level tracing tool 31 in Fig. 3 performs handling said branch tracing interrupt) in response to receiving the signal at the interrupt unit (See Fig. 3 and col. 4, lines 12+).

Referring to claim 21, Smolders teaches the second sub-instructions (i.e., instructions for counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) including

- instructions for executing code associated with the interrupt (i.e., instructions for performing interrupt handling for branch tracing interrupt; See col. 3, lines 55-61, wherein in fact, the interrupt handler code starts executing and the instruction flow unit is programmed to generate a trace interrupt after each branch inherently anticipates the claimed limitation "instructions for executing code associated with the interrupt").

Referring to claim 22, Smolders teaches

- the code records cache misses by a functional unit (i.e., load/store unit) attempting to access instructions in a cache (i.e., L2 Cache; See col. 3, lines 43-44).

Claim Rejections - 35 USC § 103

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

14. Claims 2, 11, and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Smolders [US 6,253,338 B1] as applied to claims 1, 3-6, 8-10, 12-15, 17, 18, and 20-22 above, and further in view of Torrey [US 6,145,123 A].

Referring to claim 2, Smolders discloses all the limitations of the claim 2, except that does not expressly teach responsive to receiving a subsequent instruction in the subsequent instructions indicating disablement of the mode of operation in which interrupts are to be generated, disabling the mode of operation in which interrupts are to be generated.

Torrey discloses an information processing system with trace on/off control (See Abstract), wherein

- responsive to receiving an instruction for execution in an instruction cache (i.e., Core 112 of Fig. 1) in a processor (i.e., Processor 110 of Fig. 1) being in a data processing system (i.e., information processing system 600 of Fig. 6; See col. 5, lines 9-11), determining whether the instruction indicates enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19); and
- responsive to receiving a subsequent instruction (i.e., a next instruction) in subsequent instructions (i.e., next instructions in Step 530 of Fig. 5) indicating disablement of a mode of operation (i.e., turning off trace operation; See Fig. 5, Steps 530 and 560) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19), disabling the mode of operation in which interrupts are to be generated (i.e., turning off trace operation; See Fig. 5, Step 570, and col. 10, lines 11-16).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said trace on/off control, as disclosed by Torrey, in said data processing system, as disclosed by Smolders, so as to control said mode of operation (i.e., setting/resetting BE bit in MSR for branch trace mode), for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

Referring to claim 11, Smolders discloses all the limitations of the claim 11, except that does not expressly teach disabling means, responsive to receiving a subsequent instruction in the subsequent instructions indicating disablement of the mode of operation in which interrupts are to be generated, for disabling the mode of operation in which interrupts are to be generated.

Torrey discloses an information processing system with trace on/off control (See Abstract), wherein

- enabling means (i.e., means for controlling trace on), responsive to receiving an instruction for execution in an instruction cache (i.e., Core 112 of Fig. 1) in a processor (i.e., Processor 110 of Fig. 1) being in a data processing system (i.e., information processing system 600 of Fig. 6; See col. 5, lines 9-11), determining whether the instruction indicates enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19); and
- disabling means (i.e., means for controlling trace off), responsive to receiving a subsequent instruction (i.e., a next instruction) in subsequent instructions (i.e., next instructions in Step 530 of Fig. 5) indicating disablement of a mode of operation (i.e., turning off trace operation; See Fig. 5, Steps 530 and 560) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19), for disabling the mode of operation in which interrupts are to be generated (i.e., turning off trace operation; See Fig. 5, Step 570, and col. 10, lines 11-16).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said means for controlling trace on/off, as disclosed by Torrey, in said data processing system, as disclosed by Smolders, so as to control said mode of operation (i.e., setting/resetting BE bit in MSR for branch trace mode), for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

Referring to claim 19, Smolders discloses all the limitations of the claim 19, except that does not expressly teach fourth instructions, responsive to receiving a subsequent instruction in the subsequent instructions indicating disablement of the mode of operation in which interrupts are to be generated, for disabling the mode of operation in which interrupts are to be generated.

Torrey discloses an information processing system with trace on/off control (See Abstract), wherein

- fourth instructions (i.e., means for controlling trace on), responsive to receiving an instruction for execution in an instruction cache (i.e., Core 112 of Fig. 1) in a processor (i.e., Processor 110 of Fig. 1) being in a data processing system (i.e., information processing system 600 of Fig. 6; See col. 5, lines 9-11), determining whether the instruction indicates enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19); and furthermore
- said fourth instructions (i.e., instructions for controlling trace off), responsive to receiving a subsequent instruction (i.e., a next instruction) in subsequent instructions (i.e., next instructions in Step 530 of Fig. 5) indicating disablement of a mode of operation (i.e., turning off trace operation; See Fig. 5, Steps 530 and 560) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19), for disabling the mode of operation in which interrupts are to be generated (i.e., turning off trace operation; See Fig. 5, Step 570, and col. 10, lines 11-16).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have implemented said fourth instructions for controlling trace on/off, as disclosed by Torrey, on said data processing system, as disclosed by Smolders, so as to control said mode of operation (i.e., setting/resetting BE bit in MSR for branch trace mode), for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

15. Claims 7 and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Smolders [US 6,253,338 B1] as applied to claims 1, 3-6, 8-10, 12-15, 17, 18, and 20-22 above, and further in view of Adl-Tabatabai et al. [US 6,928,582 B2; hereinafter Adl-Tabatabai].

Referring to claims 7 and 16, Smolders discloses all the limitations of the claims 7 and 16, respectively, except that does not teach the instruction is received in a bundle and wherein an indicator comprises at least one spare bit in a field in the bundle.

Adl-Tabatabai discloses a system and method for fast exception handling (See Abstract) with a 64-bit microprocessor built by Intel Corporation providing an IA-64 instruction set architecture for extensive parallel processing (See col. 3, lines 28-31), wherein

- an instruction (i.e., IA-64 instruction) is received in a bundle (i.e., 128-bit instruction bundle; See col. 3, lines 32-33) and wherein
  - an indicator (i.e., no-op instruction) comprises at least one spare bit in a field (i.e., a few of the bit space allotted) in the bundle (e.g., 2 bits to identify enabling/disabling branch tracing interrupt using said no-op instruction, which is inserted into said 128-bit instruction bundle; See col. 3, lines 39-47).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have applied said architecture of instruction bundle, as disclosed by Adl-Tabatabai, to said processor in said data processing system, as disclosed by Smolders, for the advantage of allowing multiple instructions to be fetched in each machine cycle, then the instructions are processed in some level of parallelism (See Adl-Tabatabai, col. 3, lines 31-34).

Conclusion

16. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Yeh et al. [US 6,240,510 B1] disclose system for processing a cluster of instructions where the instructions are issued to the execution units having a priority order according to a template associated with the cluster of instructions.

Puzak et al. [US 6,560,693 B1] disclose branch history guided instruction/data prefetching.

Alpert et al. [US 5,740,413 A] disclose method and apparatus for providing address breakpoints, branch breakpoints, and single stepping.

Levine et al. [US 6,067,644 A] disclose system and method monitoring instruction progress within a processor.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Christopher E. Lee whose telephone number is 571-272-3637. The examiner can normally be reached on 9:30am - 5:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Rehana Perveen can be reached on 571-272-3676. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Christopher E. Lee
Examiner
Art Unit 2112

CEL/